REMARKS

The Examiner has rejected Claims 1, 3-11, 13-19, 30, and 31-35 under 35 U.S.C. 103(a) as being unpatentable over Vaidya (U.S. Patent No. 6,279,113 B1) in view of Li et al. (U.S. Patent No. 6,567,408 B1). The Examiner has also rejected Claims 20-29 under 35 U.S.C 103(a) as being unpatentable over Copeland, III (U.S. Publication No. 2002/0144156 A1) in view of Li et al (U.S. Patent No. 6,567,408 B1). Applicant respectfully disagrees with such rejection.

With respect to each of the independent claims, the Examiner has relied on the following excerpts, along with Figure 7A, from Li to make a prior art showing of applicant's claimed technique "wherein the classification is carried out by a first classification stage capable of classifying the data packets based on a first set of packet characteristics, and a second classification stage capable of classifying the data packets received from the first classification stage based on a second set of characteristics" (see this or similar, but not necessarily identical language in each of the independent claims).

> "**The method includes providing a set of packet classification rules embodied in a plurality of lookup tables. The lookup tables include a first table containing conditions on a first parameter and one or more subsequent tables linked to the first table and containing conditions on one or more corresponding subsequent parameters.** The rules define a plurality of classes. The invention can identify a class corresponding to a packet and the class information may then be used to schedule the dispatch of the packet according to the QoS requirements of the class to which the packet belongs." (Col. 3, line 64 to Col. 4, line 6 – emphasis added)

> "Incoming packets are sorted into classes according to a policy which includes a set of rules. **For each class, the rules specify the attributes which a data packet must possess for the data packet to belong to the class.** The policy preferably also establishes QoS levels for the different classes. FIG. 4 schematically illustrates one possible policy 39. Policy 39 is specified in the form of a "policy tree" or "classification tree" for each output port of ESP 24 (an ESP 24 may have several separate output ports). The tree has a number of leaf nodes 40, 42, 44, 46. Each leaf node corresponds to a class. Each class may be treated differently in order to provide guaranteed levels of QoS to selected applications. At any given time, ESP 24 may be

> holding zero, one, or more packets belonging to each class. The packets in a class may belong to zero, one, or more flows. Non-leaf nodes of policy tree 39 may also be called "classes" although the classes into which packets are initially classified correspond to leaf classes of policy tree 39.
>
> In the example of FIG. 4, a class 40 contains voice traffic. Class 40 may be termed a "real time" class because it is important to deliver packets in class 40 quickly enough to allow a voice conversation. Packets in class 40 will be scheduled so that each flow in class 40 will be guaranteed a level of QoS sufficient for voice communication including sufficient bandwidth to support a real time voice session. Class 40 is entitled to at least 40% of the bandwidth available. The number of simultaneous flows in class 40 may be limited to a maximum value so that each flow will be guaranteed sufficient bandwidth to support a real time voice conversation." (Col. 6, lines 37-67 - emphasis added)

Applicant respectfully asserts that Li teaches classifying packets into a plurality of lookup tables based on conditions matching a single parameter. These single parameter lookup tables are then linked to form a class of packets based on rules specifying the attributes for data packets belonging to that class (see emphasized excerpts above). However, applicant claims "classifying the data packets received <u>from the first classification stage</u> based on a second set of characteristics" (emphasis added). Thus, the classes disclosed by Li are not "classifying the data packets received <u>from the first classification stage</u> based on a second set of characteristics" where the "first classification stage [is] capable of classifying the data packets based on a first set of packet characteristics," in the context claimed by applicant.

To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and the reasonable expectation of success must both be found in the prior art and not based on applicant's disclosure. *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed.Cir.1991).

Applicant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the prior art references, when combined, fail to teach or suggest all of the claim limitations, as noted above.

Applicant further notes that the prior art is also deficient with respect to the dependent claims. For example, with respect to dependent Claims 4 et al. and 6, the Examiner has relied on the following excerpts from Vaidya to meet applicant's claimed "classifying said packets within each of the groups according to packet type or size" (see Claim 4) where "classifying said packets according to packet size or type comprises classifying said packets according to packet length" (see Claim 6). Specifically, the Examiner has merely stated that Vaidya teaches "classifying said packets according to at least one packet field into groups."

> "**The attack signature profile type can be either simple, sequential or a timer/counter based.** If in step 64 the data collector 10 determines that the data packet is not associated with a network intrusion, the data collector continues to monitor data in step 58. If a network intrusion is detected, the reaction module is notified in step 66. The reaction module 38 takes steps to trace the application session associated with the data packet, to terminate the session, and/or to notify the network administrator.
>
> With reference to FIG. 4, the operation of the virtual processor 36 includes monitoring network data 46 to determine whether the data is associated with a network intrusion. A register cache 40 temporarily stores information extracted from a data packet which determines which signature profile(s) will be accessed from the signature profile memory 39. The virtual processor 36 obtains a data packet from a queue and extracts MAC header information, IP header information, transport header information, and application information from the data packet. Extraction of the packet information enables the data collector 10 to detect network intrusions based in the different layers of the OSI model." (Col. 7, lines 2-21 - emphasis added)
>
> "With reference to FIG. 8, an attack signature profile 198 can be represented as at least one expression 194 in combination with a signature attribute 196, wherein the expressions can be composed of search primitives 188, value primitives 190, and operators 192. In a preferred mode, the expressions also include keywords 193. An example of an expression might be as follows: (IP AND S1 and (V1>200)), wherein "IP" is a keyword referring to a packet utilizing IP/TCP protocol, "S1" is a search primitive referring

> to user A, "AND" is a conjunctive operator, ">200" is an operator for indicating a value greater than 200, and "V1" is a value primitive referring to a packet length. **Taken together, the entire expression describes a data packet which utilizes IP/TCP protocol, has a source address of user A and which has a packet length of greater than 200 bits.** (Col. 9, lines 46-61 - emphasis added)

Applicant respectfully asserts that the first excerpt from Vaidya cited above merely teaches attack signature profiles that may be simple, sequential, or timer/counter based (see emphasized excerpts above). These signature profiles are expressions that describe matching data packets. Nowhere within the teachings for the attack signature profiles is there any disclosure that the packets are "classified ... within each of the groups," in the manner specifically claimed by applicant. The signature profiles taught by Vaidya apply to packets meeting the signature profile criteria and do not meet applicant's claimed "classifying said packets <u>within each of the groups</u> according to <u>packet type or size</u>" (see Claim 4 et al.-emphasis added), let alone where "classifying said packets according to packet size or type comprises classifying said packets according to packet length" (see Claim 6).

With respect to dependent Claim 14, the Examiner has relied upon Col. 7, lines 2-11 and Col. 9, lines 27-35 in Vaidya, as excerpted below, to make a prior art showing of applicant's claimed technique "wherein the lookup is performed in a flow table and further comprising updating a field of the flow table." Specifically, the Examiner has stated that Vaidya "teaches performing a table lookup to select an action to be performed on said packet based on it classification."

> "**The attack signature profile type can be either simple, sequential or a timer/counter based.** If in step 64 the data collector 10 determines that the data packet is not associated with a network intrusion, the data collector continues to monitor data in step 58. If a network intrusion is detected, the reaction module is notified in step 66. The reaction module 38 takes steps to trace the application session associated with the data packet, to terminate the session, and/or to notify the network administrator." (Col. 7, lines 2-11)

> "Referring to FIG. 7, **a method for building the instruction cache 42 includes the step 112 of creating a hash index based on the**

> **server IP address and the application information in the register** cache 40. Alternatively, if the network object being monitored is a workstation, the hash index can be created using an IP address of the workstation. In step 114 **the hash index is used to search the signature profile memory 39 for a set of attack signature profiles corresponding to the server and application associated with the packet information in the register cache 40.**" (Col. 9, lines 27-35 - emphasis added)

Applicant respectfully asserts that Vaidya discloses a method creating a hash index based on a server IP address and application information in a register cache (see emphasized excerpt above). The hash index is used to search the signature profile memory for a set of attack signature profiles for associated packet information. Vaidya only generally describes the use of a hash index and does not specifically disclose the technique where a "lookup is performed in a <u>flow table</u>" or "<u>updating</u> a field of the flow table," as specifically claimed by applicant (emphasis added).

Again, applicant respectfully asserts that at least the third element of the *prima facie* case of obviousness has not been met, since the prior art references, when combined, fail to teach or suggest <u>all</u> of the claim limitations, as noted above. Thus, a notice of allowance or specific prior art showing of each of the foregoing claim elements, in combination with the remaining claimed features, is respectfully requested.

Thus, all of the independent claims are deemed allowable. Moreover, the remaining dependent claims are further deemed allowable, in view of their dependence on such independent claims.

In the event a telephone conversation would expedite the prosecution of this application, the Examiner may reach the undersigned at (408) 505-5100. The Commissioner is authorized to charge any additional fees or credit any overpayment to Deposit Account No. 50-1351 (Order No. NAI1P318/01.240.01).

Respectfully submitted,
Zilka-Kotab, PC.

Kevin J. Zilka
Registration No. 41,429

P.O. Box 721120
San Jose, CA 95172-1120
408-505-5100